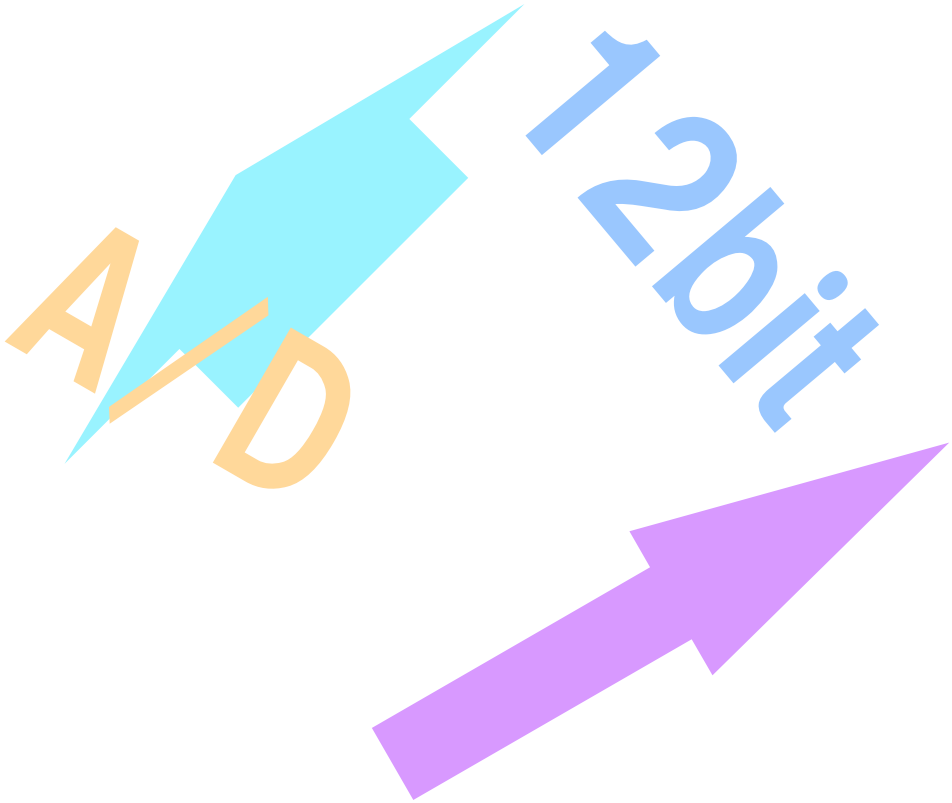
TUSB-1612ADSM

USBインタフェース付きADコンバータユニット

取扱い説明書



本文中のマークについて(必ず始めにお読み下さい)

この取扱説明書には、あなたや他の人々への危害や財産への損害を未然に防ぎ、本製品を安全にお使いいただくために、守っていただきたい事項を示しています。
その表示と図記号の意味は次のようになっています。内容をよみ理解してから本文をお読み下さい。

⚠ 警告

この表示を無視して、誤った取扱をすると、人が死亡または重傷を負う可能性がある内容を示しています。

⚠ 注意

この表示を無視して、誤った取扱をすると、人が損害を負う可能性が想定される内容および物的損害のみの発生が想定される内容を示しています。

① 製品の仕様および取扱説明書の内容は予告なく変更することがあります。
② 本製品および本取扱説明書の一部または全部を無断転載することは禁じられています。
③ 本取扱説明書の内容は万全を期して作成いたしましたが、万が一ご不審な事やお気づきの事がございましたら、（株）タートル工業　サービス課までご連絡下さい。
④ 当社では、本製品の運用を理由とする損失、逸失利益等の請求につきましては、上記に関わらずいかなる責任も負いかねますので、予めご了承下さい。
⑤ 本製品は、人命に関わる設備や機器、高度な信頼性を必要とする設備や機器などへの組込や制御などへの使用は意図されておりません。これら設備や機器などに本装置を使用され人身事故、財産損害などが生じても、当社はいかなる責任も負いかねます。
⑥ 本製品およびソフトウェアが外国為替及び外国貿易管理法の規定により戦略物資（又は役務）に該当する場合には日本国外へ輸出する際に日本国政府の輸出許可が必要です。

©2000-2005 Turtle Industry Co., Ltd. All rights reserved.
株式会社タートル工業の許可なく、本書の内容の複製、改変などを行うことはできません。

Microsoft, Windows, Windows NT, は、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
その他、記載されている会社名、製品名は、各社の商標および登録商標です。

使用上の警告と注意

⚠ 警告

入出力端子に仕様に規定された信号以上の高電圧をかけないで下さい。高電圧をかけると感電の危険性と装置破損の可能性があります。

電源アダプタは指定の物をご使用下さい。誤った電源を入力すると感電の危険性と装置破損の可能性があります。

水や薬品のかかる可能性のある場所でご使用ならさないでください。火災やその他の災害の原因となる可能性があります。

発火性ガスの存在するところでご使用なさらないでください。引火により火災、爆発の可能性があります。

煙や異臭の発生した時は直ちにご使用をおやめ下さい。ACアダプタおよびUSBケーブルを取り外し、当社サービス課までご相談下さい。

⚠ 注意

温度の高い場所では使用しないでください。故障や火災の原因となります。

不安定な所には設置しないでください。落下によりけがをする恐れがあります。

腐食性のあるガスの存在するところでは使用しないで下さい。故障や火災の原因となります。

目次

1　はじめに ―― 1
1.1　製品概要
1.2　製品構成
2　各部の名称 ―― 2
2.1　フロントパネル
2.2　リアパネル
2.3　内部
3　各端子説明 ―― 3
3.1　入出力コネクタ
3.2　電源入力コネクタ
3.3　USBコネクタ
4　インストールの方法 ―― 5
4.1　サンプルソフトのインストールと使い方
5　機能説明 ―― 6
5.1　ADコンバータの機能
5.2　ディジタル入出力の機能
5.3　AD変換値と電圧の関係
6　ドライバソフトウェアの使用 ―― 8
6.1　開発環境の設定
6.2　基本的な関数使用の流れ
7　ドライバ関数リファレンス ―― 10
8　その他 ―― 21
8.1　うまく動作しないとき
8.2　USBについて
8.3　連絡先
9　仕様 ―― 24

1 はじめに

この度は、（株）タートル工業製のUSBインターフェース付きADコンバータユニットTUSB-1612ADSMをお買い求めいただき、誠にありがとうございます。
本書は、本製品の特徴、使用方法、取扱における注意事項、その他本製品に関する情報など、本製品をご使用される上で必要な事項について記述されております。
本製品の使用には製品の性質上、若干の電子回路の知識を必要とします。誤った使用をすると本製品の破損だけでなく重大な事故が発生する事も考えられます。本書の内容をよくご理解の上、正しくご使用下さる様お願いします。

1.1 製品概要

本製品は、先進のインタフェースであるUSB(Universal Serial Bus)を使用したAD(アナログ→ディジタル)コンバータユニットです。32Kサンプルのメモリを内蔵し、計測制御において幅広い応用が可能です。ドライバソフトウェアおよびVisual C++ 6.0とVisual Basic 6.0のサンプルソフトウェアを付属しておりますので、これらの応用によって短時間に利用する事が可能です。

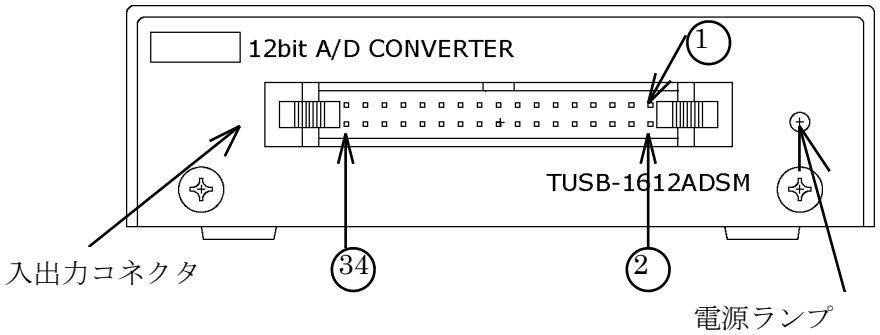
1.2 製品構成

本製品には以下の物が含まれます。
① TUSB-1612ADSM本体
② USBケーブル(1m)
④ 添付ソフトウェア(CD-ROM 1枚)

不足品などがあれば、当社サービス課までご連絡下さい。

2 各部の名称

2.1 フロントパネル

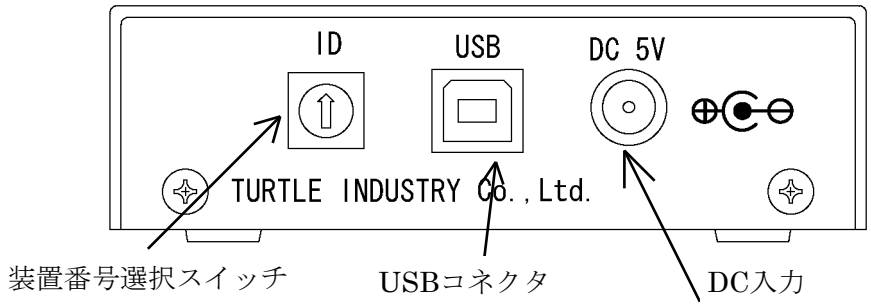


入出力コネクタ　：　フラットケーブル用34ピンコネクタ(ロック付き)です。ADコンバータ、ディジタル入出力の入出力をここから行います。

電源ランプ　　：　電源ON時に点灯します。

※　○で囲まれた数字はコネクタのピン番号です。

2.2 リアパネル



USBコネクタ　：　コンピュータと付属のUSBケーブルで接続します

DC入力　　：　外部電源使用時に専用電源を接続します

ユニット番号選択(ID):　本ユニットのユニット番号を選択します

3 各端子説明

3.1 入出力コネクタ

ADコンバータ、ディジタル入出力はフロントパネルの入出力コネクタを使用します。

ケーブル側コネクタ：ヒロセ電機（株）製 HIF3BA-34D-2.54Rまたは同等品

ピン番号	シングルエンド		ディファレンシャル(差動)	
	名称	機能	名称	機能
1	AD0	チャンネル1入力	AD0+	チャンネル1入力＋
2	AD1	チャンネル2入力	AD0-	チャンネル1入力－
3	GND	信号グラウンド	GND	信号グラウンド
4	AD2	チャンネル3入力	AD1+	チャンネル2入力＋
5	AD3	チャンネル4入力	AD1-	チャンネル2入力－
6	GND	信号グラウンド	GND	信号グラウンド
7	AD4	チャンネル5入力	AD2+	チャンネル3入力＋
8	AD5	チャンネル6入力	AD2-	チャンネル3入力－
9	GND	信号グラウンド	GND	信号グラウンド
10	AD6	チャンネル7入力	AD3+	チャンネル4入力＋
11	AD7	チャンネル8入力	AD3-	チャンネル4入力－
12	GND	信号グラウンド	GND	信号グラウンド
13	AD8	チャンネル9入力	AD4+	チャンネル5入力＋
14	AD9	チャンネル10入力	AD4-	チャンネル5入力－
15	GND	信号グラウンド	GND	信号グラウンド
16	AD10	チャンネル11入力	AD5+	チャンネル6入力＋
17	AD11	チャンネル12入力	AD5-	チャンネル6入力－
18	GND	信号グラウンド	GND	信号グラウンド
19	AD12	チャンネル13入力	AD6+	チャンネル7入力＋
20	AD13	チャンネル14入力	AD6-	チャンネル7入力－
21	GND	信号グラウンド	GND	信号グラウンド
22	AD14	チャンネル15入力	AD7+	チャンネル8入力＋
23	AD15	チャンネル16入力	AD7-	チャンネル8入力－
24	GND	信号グラウンド	GND	信号グラウンド
25	PI0	ディジタル入力0 外部クロックおよび開始信号と併用		
26	GND	信号グラウンド		
27	PI1	ディジタル入力1		
28	GND	信号グラウンド		
29	PO0	ディジタル出力0		
30	GND	信号グラウンド		
31	PO1	ディジタル出力1		
32	GND	信号グラウンド		
33	+5V	5V出力（外部補助電源使用時のみ有効）		
34	GND	信号グラウンド		

※　5V電源出力はオプションの外部補助電源を使用した時のみ有効となります。
※　ディジタル入出力はTTLレベルです。

3.2　電源入力コネクタ

本ユニットはUSBバスから供給されるDC5V電源で動作します。ただし、以下の様な場合があります。必要に応じて外部電源を使用してください。

1)　コンピュータがサスペンド状態になるとUSBに供給される電源が遮断される可能性があります。
2)　サスペンド状態で電源が遮断されなくとも、USB機器の使用出来る電源電流はサスペンド状態では500μAにまで制限されます。この時には低消費電力状態で待機します。低消費電力状態ではディジタル出力は全てハイインピーダンスになるため、構成によっては本ユニットまたは相手接続装置に動作異常や故障の発生する可能性があります。
3)　ハブには自己電源をもつセルフパワードハブと自己電源をもたないバスパワードハブがあります。後者の場合は内部に電源を持たないためUSBラインから電源をとることになります。USBラインから供給される電源の電流は最小では100mAまでしか利用ません。
4)　3)の場合やその他コンピュータの構成によっては電源容量が不足する場合があります。その場合は外部補助電源を使用するか、自己電源を内蔵するハブを使用してください。

外部電源は安定化されたDC5V電源が必要となります。外部補助電源を使用される場合には専用ACアダプタ(別売)をご利用下さい。

3.3　USBコネクタ

付属のUSBケーブルを使用して、ご利用されるコンピュータまたはハブに接続してください。
※　初めて接続される時にはインストール作業が必要です。

4 インストールの方法

4.1 サンプルソフトのインストールと使い方

サンプルソフトはプロジェクトソースと共に以下の場所に格納されておりますので、適切な場所にコピーしてご使用下さい。

```
<付属ディスク>
  |-[TUSBADSM]
      |-[DRV]          : ドライバ
      |-[DOC]          : ドキュメント
      |-[DEV]
          |-[VB6]      : Visual Basic 6.0用サンプル
          |            : プロジェクト
          |-[VBNET]    : Visual Basic .NET用サンプル
          |            : プロジェクト
          |-[VC]       : Visual C++ 用サンプル
          |            : プロジェクト
          |-[TOOLS]    :DLL,LIB,H,BAS,VBファイル
```

Visual C++用のサンプルソフトとVisual Basic用のサンプルソフトの動作は同様です。Visual Basic 6.0のプロジェクトフォルダの中には構築後の実行ファイルが入っており、そのまま実行する事が出来ます。(ただしVisual Basic 6.0のランタイムライブラリが必要です。)
Visual C++のプロジェクトフォルダの中には実行ファイルはございませんので使用する場合には開発ツールで構築してください。

各アプリケーションとも起動後、本体装置のID番号を選択し、Openボタンを押下するとかくコマンドが実行できます。ボタン名にはドライバ関数の名前と同様の名前が付いております。各ボタンを押下する事により各関数の実行を確認する事が出来ます。

詳しくは各プロジェクトのソースコードをご覧下さい。

5 機能説明

5.1 ADコンバータの機能

シングル取込み
ADコンバータ16チャンネル(差動では8チャンネル)の内、指定したチャンネルから1サンプルデータを取り込む事が出来ます。連続取込み中にこの機能を使用する事は出来ません。

変換範囲
シングルエンドタイプでは入力範囲をバイポーラとユニポーラから選択する事が出来ます。入力チャンネル毎に設定する事は出来ません。

内部トリガ連続取込み
内部タイミング信号生成器を使用して連続的に波形をサンプリングする事ができます。
○　最大サンプリングレートは100KHzです。
○　連続取込をディジタル入力0の立ち上がりでスタートする事も出来ます。

外部トリガ取込み
ディジタル入力0の立ち上がりに連動してサンプリングをする事が出来ます。
○　取込間隔は10μ秒以上にしてください。
○　取込間隔は一定周期で無くとも取込む事が可能です。
○　予め指定したメモリ量に達した時点で取込は停止します。

DCカットされた出力の接続ついて
機器によっては出力にコンデンサが入っており、DCレベルが確定していない場合があります。この様な時な機器を接続した場合には正確な値をとる事が出来ません。出力を抵抗器でグラウンドに接続するなど、DCレベルが±10V以内となる様な処理をして使用してください。

⚠ 注意

入力端子に±12Vを越える電圧をかけると装置の破損または、性能が劣化する可能性があります。

5.2 ディジタルIOの機能

ロジックレベル
ディジタル入出力の論理レベルは全てTTLレベルです。

出力のシンク、ソース電流
ソース電流、シンク電流とも2mAです。

ADコンバータトリガ入力
ディジタル入力0はADコンバータのサンプリングトリガおよび連続取込開始トリガと併用です。

5.3 AD変換値と電圧の関係

AD変換値と入力電圧の関係は次のようになっています。

レンジ	変換値		
	0	2048	4095
±2.5V	-2.5V	0V	+2.499V
0～5V	0V	2.5V	4.999V
±5V	-5V	0V	+4.998V
0～10V	0V	5V	9.998V

※ 12ビット変換値の最大はFFF(HEX)となり、変換レンジの最大値より1LSB小さくなっています。
※ 上記表は誤差は含まれておりません。

変換値から電圧への変換式(例)は次のようになります。

レンジ	式(変換値をA、電圧をVとする)
±2.5V	V＝A×5.0÷4096.0－2.5
0～5V	V＝A×5.0÷4096.0
±5V	V＝A×10.0÷4096.0－5.0
0～10V	V＝A×10.0÷4096.0

6 ドライバソフトウェアの使用

6.1 開発環境の設定

Visual C++の場合

1 付属のフロッピーディスクより(TOOLSディレクトリの中)
USBADSM.LIB
TUSBADSM.H
を適当な場所にコピーします。

2 USBADSM.LIBファイルをプロジェクトに追加します。

3 使用するソースファイルにTUSBADSM.Hファイルをインクルードします。

設定は以上です。
本サンプルプログラムはVisual C++ 6.0で作成されました。
※ Visual C++ .NET でも使用することが出来ます。

Visual Basic 6.0 の場合

1 付属のフロッピーディスクより(TOOLSディレクトリの中)
TUSBADSM.BAS
を適当な場所にコピーします。

2 TUSBADSM.BASファイルをプロジェクトに追加します。

設定は以上です。

Visual Basic .NET の場合

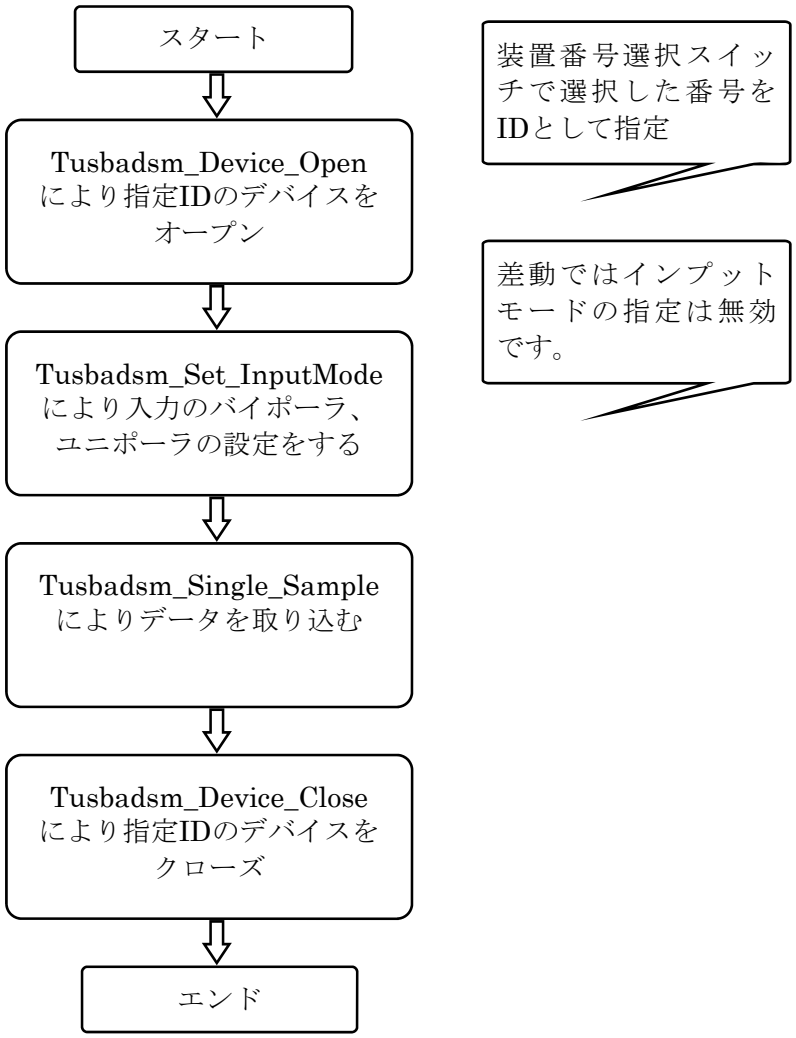
1 付属のフロッピーディスクより(TOOLSディレクトリの中)
TUSBADSM.vb
を適当な場所にコピーします。

2 TUSBADSM.vbファイルをプロジェクトに追加します。

設定は以上です。

6.2　基本的な関数使用の流れ

ここでは、関数の使用方法を簡単な例を元にご説明いたします。この関数はデバイスをオープンし、入力モードを設定して変換データを一つ取得します。



オープンおよびクローズはプログラムの開始時および終了時に一回ずつ行う必要があります。一回の作業後毎にオープン、クローズを行う必要はありません。
その他の関数については関数リファレンスをご参照下さい。
２台以上使用する場合には装置の選択番号を変えて、それぞれについてオープンクローズを行って下さい。

7 ドライバ関数リファレンス

Tusbadsm_Device_Open

C,C++宣言	short __stdcall Tusbadsm_Device_Open(short id)
VB6定義	[Private] Declare Function Tusbadsm_Device_Open Lib "USBADSM.DLL" (ByVal id As Integer) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Device_Open Lib "USBADSM.DLL" (ByVal id As Short) As Short

解説

指定ID(ユニット番号選択スイッチの値)のデバイスをオープンします。
このデバイスに関する各種関数を使用する前に必ず呼び出す必要が有ります。

引数

id	ユニット番号選択スイッチの番号(0から15)

戻り値

0:成功
1:ID番号が不正
2:ドライバがインストールされていない
3:デバイスはすでにオープンされている
4:接続されている台数が多すぎる(最高16台まで)
5:オープンできなかった
6:デバイスが見つからない

Tusbadsm_Device_Close

C,C++宣言	void __stdcall Tusbadsm_Device_Close(short id)
VB6定義	[Private] Declare Sub Tusbadsm_Device_Close Lib "USBADSM.DLL" (ByVal id As Integer)
VB.NET定義	[Private] Declare Sub Tusbadsm_Device_Close Lib "USBADSM.DLL" (ByVal id As Short)

解説

指定ID(ユニット番号選択スイッチの値)のデバイスをクローズします。

引数

id	ユニット番号選択スイッチの番号(0から15)

戻り値

なし

Tusbadsm_Pio_Write

C,C++宣言	short __stdcall Tusbadsm_Pio_Write (short id, unsigned char dat);
VB6定義	[Private] Declare Function Tusbadsm_Pio_Write Lib “USBADSM.DLL” (ByVal id As Integer , ByVal dat As Byte) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Pio_Write Lib “USBADSM.DLL” (ByVal id As Short , ByVal dat As Byte) As Short

解説

ディジタル出力の出力データを指定します。指定方法はビットパターンで行い、例えば引数datに3を指定すれば全てHigh、0を指定すれば全てLowとなります。

引数

id	ユニット番号選択スイッチの番号(0から15)
dat	出力データの指定(0～3)

戻り値

0:成功
1:オープンされていない
2:失敗

Tusbadsm_Pio_Read

C,C++宣言	short __stdcall Tusbadsm_Pio_Read (short id, unsigned char *dat);
VB6定義	[Private] Declare Function Tusbadsm_Pio_Read Lib “USBADSM.DLL” (ByVal id As Integer , ByRef dat As Byte) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Pio_Read Lib “USBADSM.DLL” (ByVal id As Short , ByRef dat As Byte) As Short

解説

ディジタル入力の入力データを読込みます。入力値はビットパターンで参照変数datに入力されて戻されます。引数datが3であれば入力が全てHigh、0であれば全てLowとなります。

引数

id	ユニット番号選択スイッチの番号(0から15)
dat	入力データを格納するバッファ

戻り値

0:成功
1:オープンされていない
2:失敗

Tusbadsm_Single_Sample

C,C++宣言	short __stdcall Tusbadsm_Single_Sample (short id, int ch,int *dat)
VB6定義	[Private] Declare Function Adapio_Adc_SingleSample Lib “USBADAPIO.DLL” (ByVal id As Integer , ByVal ch As Long, ByRef dat As Long) As Integer
VB.NET定義	[Private] Declare Function Adapio_Adc_SingleSample Lib “USBADAPIO.DLL” (ByVal id As Short , ByVal ch As Integer, ByRef dat As Integer) As Short

解説

ADコンバータの変換値を取得します。連続サンプリング実行時にこの関数を使用する事は出来ません。入力値はストレートバイナリです。入力範囲の最小値が0、最大値がFFF(HEX)となります。

引数

id	ユニット番号選択スイッチの番号(0から15)
ch	入力するチャンネル0～15
dat	入力データ　000(HEX)～FFF(HEX)

戻り値

0:成功
1:オープンされていない
2:失敗
3:チャンネル指定が不正か連続取込中につき実行不可

Tusbadsm_Start_Adc

C,C++宣言	short __stdcall Tusbadsm_Start_Adc (short id,int num,int trg1,int trg2,int sel)
VB6定義	[Private] Declare Function Tusbadsm_Start_Adc Lib “USBADSM.DLL” (ByVal id As Integer ,ByVal num As Long, ByVal trg1 As Long , ByVal trg2 As Long, ByVal sel As Long) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Start_Adc Lib “USBADSM.DLL” (ByVal id As Short ,ByVal num As Integer, ByVal trg1 As Integer, ByVal trg2 As Integer, ByVal sel As Integer) As Short

解説

内部または外部トリガに連動してメモリに連続的に変換データを記録します。
内部トリガを使用する場合取込間隔はtrg1×trg2(μ秒)となります。この値は必ず10μ秒以上として下さい。内部トリガはこの関数の実行後すぐに設定された間隔でデータを取込ます。
外部トリガはディジタル入力0の立ち上がりに連動してデータをサンプリングします。
外部開始内部トリガはディジタル入力0の立ち上がりでスタートし、内部トリガ間隔でサンプリングをします。
この関数の実行に先立ってTusbadsm_Set_Pattern関数の実行が必用です。

引数

id	ユニット番号選択スイッチの番号(0から15)
num	取込データ数（1～32768)
trg1	サンプルトリガ設定1(1～65536)
trg2	サンプルトリガ設定2(1～65536)
sel	トリガ選択　0:内部　1:外部　2:外部開始内部トリガ

戻り値

0:成功
1:オープンされていない
2:失敗
3:設定値が不正または連続取込中につき開始出来ない

Tusbadsm_Stop_Adc

C,C++宣言	short __stdcall Tusbadsm_Stop_Adc (short id)
VB6定義	[Private] Declare Function Tusbadsm_Stop_Adc Lib “USBADSM.DLL” (ByVal id As Integer) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Stop_Adc Lib “USBADSM.DLL” (ByVal id As Short) As Short

解説

ADコンバータの連続取込動作を中止します。

引数

id	ユニット番号選択スイッチの番号(0から15)

戻り値

0:成功
1:オープンされていない
2:失敗

Tusbadsm_Get_Status

C,C++宣言	short __stdcall Tusbadsm_Get_Status (short id,int *st)
VB6定義	[Private] Declare Function Tusbadsm_Get_Status Lib “USBADSM.DLL” (ByVal id As Integer , ByRef st As Long) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Get_Status Lib “USBADSM.DLL” (ByVal id As Short , ByRef st As Integer) As Short

解説

連続サンプリング動作のステータスを取得します。
stが0になるとサンプリング終了です。
Tusbadsm_Start_Adcの実行後でないと値には意味がありません。

引数

id	ユニット番号選択スイッチの番号(0から15)
st	サンプリング残データ数

戻り値

0:成功
1:オープンされていない
2:失敗

Tusbadsm_Get_Datas

C,C++宣言	short __stdcall Tusbadsm_Get_Datas (short id,int num, int *dat)
VB6定義	[Private] Declare Function Tusbadsm_Get_Datas Lib “USBADSM.DLL” (ByVal id As Integer , ByVal num As Long, ByRef dat As Long) As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Get_Datas Lib “USBADSM.DLL” (ByVal id As Short , ByVal num As Integer, ByRef dat As Integer) As Short

解説

連続サンプリング済みデータバッファの内容を取得します。
先頭から指定された数のデータをdatに格納します。

引数

id	ユニット番号選択スイッチの番号(0から15)
num	取得するデータ数
dat	取得したデータを格納する配列の先頭

戻り値

0:成功
1:オープンされていない
2:失敗

Tusbadsm_Set_Pattern

C,C++宣言	short __stdcall Tusbadsm_Set_Pattern (short id,int num, int *st, int len)
VB6定義	[Private] Declare Function Tusbadsm_Set_Pattern Lib “USBADSM.DLL” (ByVal id As Integer ,ByVal num As Long, ByRef st As Long,ByVal len As Long)As Integer
VB.NET定義	[Private] Declare Function Tusbadsm_Set_Pattern Lib “USBADSM.DLL” (ByVal id As Short ,ByVal num As Integer, ByRef st As Integer,ByVal len As Integer)

解説

連続取込時のチャンネルの取込順を設定します。最高16回のチャンネルパターンの繰り返しで取込ます。チャンネルの指定は配列stに0～15のチャンネル番号を順に入力して置きます。
たとえば、4ch,2ch,3ch,10chの順番に連続して各々100データ分取得するならば、num = 4, st[0] = 3 , st[1] = 1 , st[2] = 2 , st[3] = 9 ,len = 400を設定します。

○ 取込結果はメモリの中に指定したチャンネルの順に入っています。
○ この関数を実行するとメモリ内の取得済みデータは消去されます。
○ ADを連続取込でスタートする前には毎回必ずこの関数を実行しなければなりません。(取込パターンの記憶領域と取得データの領域は共有されています。)

引数

id	ユニット番号選択スイッチの番号(0から15)
num	Stパターンバッファの長さ(繰り返し長さ1～16)
st	チャンネルパターンを格納したバッファ
len	取込データ長(1～32768)

戻り値

0:成功
1:オープンされていない
2:失敗
3:設定値が不正です

Tusbadsm_Set_InputMode

C,C++宣言	short __stdcall Tusbadsm_Set_InputMode (short id, int mode)
BASIC定義	[Private] Declare Function Tusbadsm_Set_Pattern Lib “USBADSM.DLL” (ByVal id As Integer ,ByVal mode As Long) As Integer
BASIC定義	[Private] Declare Function Tusbadsm_Set_Pattern Lib “USBADSM.DLL” (ByVal id As Short ,ByVal mode As Integer) As Short

解説

入力モードを設定します。
○　設定は全チャンネルに反映されます。
○　ディファレンシャル(差動)型では設定は無効です。

引数

id	ユニット番号選択スイッチの番号(0から15)
mode	入力モード 0:バイポーラ　1:ユニポーラ

戻り値

0:成功
1:オープンされていない
2:失敗

8 その他

8.1 うまく動作しないとき

ユニットが認識(インストール)できない
OSはWindows 98 、Meまたは Windows 2000、XPですか ———>その他のOSには対応しておりません。 電源ランプが点灯していない時 ———>USBケーブルを差直してください。

デバイスがオープンできない
デバイスマネージャなどでデバイスが正しくインストールされていることを確認してください。正しくインストールされていない場合は、ドライバのインストールを行って下さい。

一度はデバイスがオープンできるが再オープンできない
デバイスを一度オープンしたらクローズするまで再オープンできません。デバイスの使用が終了するかまたはアプリケーション終了時に確実にクローズしてください。

8.2 USBについて

USBとはUniversal Serial Busの頭文字の略で、新しいコンピュータのインターフェースバスです。インターフェースのコストが低く使い易い事などからパーソナルコンピュータを中心に普及しました。USB1.1の仕様では、1.5Mbpsロースピードデバイスおよび12Mbpsハイスピードデバイスがあります。本ユニットでは12Mbpsハイスピード仕様になっております。

USBの主な特長	
高速	最高12Mbpsで通信可能(USB2.0では480Mbps)
接続が容易	ISAやPCIなどの拡張バスと違いケーブル1本で接続可能。コンピュータの動作中でも抜き差し可能。
多数接続可能	ハブの利用により最高127台(ハブを含む)のデバイスを接続可能。
電源の供給	標準で100mA、最大で500mAの電源をバスで供給可能。
低コスト	多くのパーソナルコンピュータに標準で装備されており、安価なケーブル1本で接続可能。ただし、標準装備のポート数より多くのデバイスを接続する際にはハブが必要。

ハブについて

多数のUSBを接続するにはハブデバイスが必要です。ハブは1本のUSB線(上流側)を複数のUSB線(下流側)に分岐します。ハブにはバスパワードハブとセルフパワードハブがあり、前者は上流側の電源により動作しますが、後者は外部電源により動作します。ホストのポートからは標準で100mA、最大500mAの電流を供給する事が出来ます。バスパワードハブでは通常100mA未満の電流を消費するため、このハブに接続されたデバイスはバスから500mAを供給される事は出来ません。100mA以上の電流を消費するデバイスをバスパワードハブに接続する場合には注意が必要です。

ケーブルについて

USBケーブルはAタイプとBタイプに分かれます。ホストのポートはAタイプ、デバイス側はBタイプとなっており、誤挿入が起こらない仕様になっております。

転送速度について

USBの転送速度はきわめて高速ですが、接続されたデバイスの単位時間当たりのデータ転送量の総合計が最高転送量を超える事はありません。あるデバイスで大量のデータ転送を行うと他のデバイスの転送速度に影響の出る可能性があります。

8.3　連絡先

動作上の問題点および不明な点などのお問い合わせは下記までお願いします。
調査の上、当社よりご連絡差し上げます。

ご質問の際には動作環境等、なるべく詳細な情報を下さい。
特に次の情報は必ず記載してください。

ご使用のコンピュータの機種
ご使用OS(Windows98 , Windows98 SEなど)
メモリ容量
ハードディスクの容量
本ユニット以外でご使用されているUSB装置
こちらからご連絡差し上げる場合の貴ご連絡先

株式会社タートル工業
～　技術部　技術課　サービス係　～

E-mail	info@turtle-ind.co.jp
FAX	0298-43-2024
郵送	〒300-0842 茨城県土浦市西根南1-12-4

9 仕様

ADコンバータ部

チャンネル数	差動8ch又はシングルエンド16ch
チャンネル切換時間	5μ秒(フルスイング90%以上)
変換分解能	1/4096(12bit)
変換誤差	±4LSB
変換ビットパターン	ストレートバイナリ
変換レート	100KHz(MAX)
メモリ容量	12bit×32Kサンプル
入力電圧範囲	機種選択表参照
絶対最大入力電圧	＋12V
絶対最小入力電圧	－12V
変換トリガ	内部クロックまたは外部TTL
入力抵抗	約1MΩ

ディジタル入出力部

ビット数	出力2bit/入力2bit(外部トリガ兼用)
出力レベル	TTL(出力は2mA シンク, ソース)

その他

信号入出力コネクタ	34Pフラットケーブル用(各部共通)
電源	USBまたは外部安定化DC
消費電流	約200mA
外部電源電圧	5.0VDC安定化されたもの
電源出力	5VDC 500mA(オプションの補助電源使用時のみ)
大きさ	30(h)×100(w)×140(d)mm(突起物含まず)
重さ	約350g(ケーブル含まず)

機種選択表

	入力方式	入力電圧	同相入力電圧
1612ADSM-1	8ch差動	±2.5V	±10V
1612ADSM-2	16chシングルエンド	0～5V,±2.5V切替	
1612ADSM-3	8ch差動	±5V	±10V
1612ADSM-4	16chシングルエンド	0～10V,±5V切替	

TUSB-1612ADSM取扱説明書

発行年月　　2008年7月　第16版

発　　行　　株式会社　タートル工業
編　　集　　株式会社　タートル工業

©2008　株式会社　タートル工業